Optimization Letters (2022) 16:515–544
https://doi.org/10.1007/s11590-021-01733-0

ORIGINAL PAPER

Minimum budget for misinformation detection in online social networks with provable guarantees

Canh V. Pham[1] · Dung V. Pham[2] · Bao Q. Bui[2] · Anh V. Nguyen[3]

Received: 20 October 2019 / Accepted: 2 April 2021 / Published online: 13 April 2021

Abstract

Misinformation detection in Online Social Networks has recently become a critical topic due to its important role in restraining misinformation. Recent studies have showed that machine learning methods can be used to detect misinformation/fake news/rumors by detecting user's behaviour. However, we can not implement this strategy for all users on a social network due to the limitation of budget. Therefore, it is critical to optimize the monitor/sensor placement to effectively detect misinformation. In this paper, we investigate *Minimum Budget for Misinformation Detection* problem which aims to find the smallest set of nodes to place monitors in a social network so that detection function is at least a given threshold. Beside showing the inapproximability of the problem under the well-known Independent Cascade diffusion model, we then propose three approximation algorithms including: Greedy, Sampling-based Misinformation Detection and Importance Sampling-based Misinformation Detection. Greedy is a deterministic approximation algorithm which utilizes the properties of monotone and submodular of the detection function. The rest is two randomized algorithms with provable guarantees based on developing two novel techniques (1) estimating detection function by using the concepts of influence sample and importance influence sample with proof of correctness, and (2) an algorithmic framework to select the solution with theoretical analysis. Experiments on real social networks show the effectiveness and scalability of our algorithms.

✉ Canh V. Pham
canh.phamvan@phenikaa-uni.edu.vn

Dung V. Pham
pvdungc500@gmail.com

Bao Q. Bui
buiquybao.c500@gmail.com

Anh V. Nguyen
anhnv@ioit.ac.vn

1 ORLab, Faculty of Computer Science, Phenikaa University, Hanoi 12116, Vietnam

2 Faculty of Information Technology and Security, People's Security Academy, Hanoi, Vietnam

3 Institute of Information Technology, Vietnam Academy of Science and Technology, Hanoi, Vietnam

Keywords Online social network · Misinformation detection · Approximation algorithms

1 Introduction

Nowadays, Online Social Networks (OSNs) have rapidly developed and become an effective platform for communication. According to recent surveys, there are about 3 billion users in OSNs and many users considered OSNs as the source of their daily information [46]. Unfortunately, OSNs are also exploited for the purpose of spreading misinformation, rumors and fake news, which have caused significant economical and political consequences, see [1,13,30]. Therefore, it is a great practical importance to effectively detect the propagation of misinformation in the social media before it causes serious consequences. This task is the motivation for several strategies to prevent misinformation such as blocking users or links [21,38,41,42] and disseminating good information to correct misinformation [4,39]. Recently, some works have shown that misinformation and fake news can be automatically detected by machine learning techniques from temporal, structural, linguistic features of users [23], content of posts and microblog-specific memes [32–34,44]. We consider all techniques used to misinformation detection to exploit user's behaviours as "monitor/sensor" placements.

Based on those studies, some authors have proposed optimal management of resources approaches to detect misinformation or outbreaks in a social network by placing the monitors/sensors at some critical nodes, such as, cascade of epidemics detection with a cost constraint [25], misinformation detection with a size of monitor-set constraint [52,53], timely misinformation detection by heuristic approaches [54], etc. However, previous works have been failed to deal with many real scenarios. Suppose that we need to monitor all the users in a group in an OSN, monitor placement strategies with the cost and size constraints are not feasible because it may not be possible to monitor all users in the group. In this scenario, a monitor placement strategy with minimal size to ensure that all nodes in the group can be monitored, is obviously more efficient than previous strategies.

Motivated by the above phenomenon, in this paper we propose the *Minimum Budget for Misinformation Detection* (MBD) problem which aims to find the smallest set of nodes to place monitors in a social network so that the detection function $\mathbb{D}(\cdot)$ which evaluates information spread from a given set of suspected misinformation node S, is at least a given threshold γ. The threshold γ can control the scale of misinformation monitoring strategy. The greater the value of γ is, the much more users are monitored. MBD is more relevant in practice as we often have to monitor misinformation throughout a network. The main challenge of this problem comes from its inapproximability and the complexity for calculating detection function. We show that the calculation of the objective function is #P-hard and it is NP-hard to approximate the problem with the ratio of $(1-\epsilon)\ln n$. To overcome this challenge, we propose two randomized algorithms with provable guarantees. Our contributions are summarized as follows:

- We formulate Minimum Budget for Misinformation Detection (MBD) under the well-known Independent Cascade (IC) information diffusion model. We show that the calculation of the objective function is #P-hard and it is NP-hard to approximate the problem with the ratio of $(1-\epsilon)\ln n$, for $\epsilon > 0$ unless NP $\in DTIME(n^{O\log\log n})$.
- We develop novel techniques to estimate function $\mathbb{D}(\cdot)$ by proposing influence sample (DS) and importance influence sample (IDS) concepts with correctness proof. Based on that, we show that $\mathbb{D}(\cdot)$ is a monotone and submodular function and propose a Greedy algorithm providing an approximation ratio of $1+\ln(\gamma/\epsilon)$ for any $\epsilon > 0$. In order to find the solution on large-scale networks, we further propose two efficient randomized algorithms, named *Sampling-based Misinformation Detection* (SMD) and *Importance Sampling-based Misinformation Detection* (ISMD), by utilizing the estimations of detection function from DS and IDS concepts and devising an algorithmic framework to select a near-optimal solution. We prove that these algorithms return a solution A satisfying $|A| \leq 1+|A^*|\cdot\ln\frac{\gamma-\gamma\epsilon}{\epsilon}$ and $\mathbb{D}(A) \geq \gamma\cdot\frac{1-\epsilon}{1+\epsilon}-\epsilon$ with high probability where $\epsilon \in (0,1)$ is an input and A^* is an optimal solution.
- We conduct extensive experiments on real social networks to demonstrate the effectiveness and scalability of our algorithms. SMD and ISMD not only give an approximation guarantee, but also can apply to very large-scale networks (Email-Eu-All network contains 265K nodes and 420K edges) and they outperform state-of-the-art algorithms in term of quality solution and running time. In addition, the results also show that ISMD needs fewer the number of required samples and memories than that of other algorithms.

Organization The rest of the paper is organized as follows. We summarize the related literature in Sect. 2. Next, we introduce the information diffusion model, problem definition and its inapproximability in Sect. 3. In Sect. 4 we present our proposed algorithms. The experiments on several datasets are in Sect. 5. Finally, we conclude the paper in Sect. 6.

2 Related works

In this section, we are going to review previous works regarding misinformation detection including: Information Diffusion models and Influence Maximization and Misinformation Detection.

Information diffusion model and influence maximization Information diffusion models is the solid background for studying information propagation issues and viral marketing. Kempe et al. [20] first propose two classical information diffusion models, named Independent Cascade (IC) and Linear Threshold (LT). Working on these models, they formulate the Influence Maximization (IM) problem which seeks k nodes that can influence to the largest number of nodes in an OSN and they devise an $(1-1/e)$ approximation algorithm for this problem. Due to great commercial values, a large number of works have focused on IM problem on proposing scalability and efficiency algorithms [3,7,8,48,49], studying IM variations [2,19,28,37,43,50]. Some

works extend the IC model by incorporating time, topic to reflect some real contexts in viral marketing. Chen et. al. [9] introduce the Independent Cascade with Meeting events model by adding the time delay aspect of influence diffusion in each link. A Continuous-time Independent Cascade model is proposed for influence estimation and maximization problems with time-sensitive context [14,17]. Several works [2,15,29] consider the topic-aware influence maximization with the purpose of maximizing influenced users with a given topic query. In this problem, each edge has multiple transmission information probabilities that reflect the influence on different topics.

Misinformation detection Misinformation, fake news and rumors can be automatically detected by using text mining techniques from sequential microblog streams [5,23,31,44]. For example, Qazninian et al. [44] study rumors identifying on the Twitter by developing three categories of features to identify the false tweets (uni gram, bigrams, and part-of-speech). Kwon et al. [23] introduce a time-series-fitting model standing on the volume of tweets over time. Ma et al. [31] capture the temporary characteristics of the time series of rumor's life-cycle for identifying rumors from online social media. More recently, several deep neural models have developed for automatic rumors detection [6,32–35,45].

The outbreak of disease occurs in many different networks. There is a common strategy to detect outbreaks from many different networks which is to place monitors or sensors into some important nodes such as water contamination [22] and detection and contagious outbreaks [10]. Besides, motivated by the fact that misinformation or rumor can be automatically detected through data mining and machine learning methods, some authors investigate the problem of placing monitor/sensor into some nodes to detect misinformation in an OSN [12,12,53,54]. Leskovec et al. investigate the problem of detection outbreaks in a blog network under the budget constraint, and devise an $(1 - 1/e)/2$ approximation algorithms for this problem [25]. Cui et al. [12] focus on selecting important nodes as sensors to predict the outbreaks with a data-driven approaching. Zhang et al. [53] investigate the problem of misinformation detection within (MD) limited budget under IC model. They show that MD problem can be viewed as IM problem when all nodes have the same probability to be a source of misinformation but they still fail to deal with the case which nodes have different probabilities to be a source. Authors in [54] focus on TCMD problem, which finds minimum-size monitor set so that misinformation can be detected from all nodes in the network within time constraint and propose a heuristic algorithm for general case. One drawback of these two studies is that the proposed algorithms do not provide any approximation guarantee.

Approaching a new view of above studies, in this work, we aim to find set of nodes with minimal size to place monitors in so that the expected detection probability is at least a threshold γ. Different from Misinformation Detection problem [53], in this task, each node u is a source of misinformation with arbitrary probability. Besides, there is no existing algorithm for our problem, we are going to propose approximation algorithms that returns near-optimal solutions with high probability.

Table 1 Table of symbols

Notional	Description
n, m	The number of nodes and the number of edges in G, respectively
$N_{in}(v)$, $N_{out}(v)$	the sets of incoming, and outgoing neighbor nodes of v
S	Set of nodes is likely to be the source of misinformation
A	Set of monitor nodes
$\mathbb{D}(A)$	Detection function of node set A
$\hat{\mathbb{D}}(A)$	An estimation of $\mathbb{D}(A)$
$\mathsf{Cov}(A, R_j)$	$= \min\{1, \lvert A \cap R_j \rvert\}$
$\mathsf{Cov}_{\mathcal{R}}(A)$	$= \sum_{R_j \in \mathcal{R}} \mathsf{Cov}(A, R_j)$, the number of DS sets in $\mathcal{R}$ covered by A
$N_i(\delta, \epsilon)$	$\frac{(2+\frac{2}{3}\epsilon)n}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(\binom{n}{i}/\delta)$

3 Model and problem definition

In this section, we introduce the network model and a well-known diffusion model Independent Cascade (IC) [20]. We then formally define the Minimum Budget for Misinformation Detection (MBD) and present the inapproximability of the problem. In Table 1, the frequently used notations are summarized.

3.1 Independent cascade model

Let $G = (V, E)$ be a directed graph representing a social network with a node set V and a directed edge set E, $|V| = n$ and $|E| = m$. Let $N_{in}(v)$ and $N_{out}(v)$ be the set of in-neighbors and out-neighbor of a node v, respectively.

In this model, each edge $e = (u, v) \in E$ has a probability $p(u, v) \in (0, 1)$ that represents the misinformation transmission from u to v. The diffusion process from S happen in discrete steps $t = 0, 1, 2 \ldots$ as follows:

- At step $t = 0$, all nodes in S are activated by the misinformation and the others are inactive.
- At step $t \geq 1$, for an activated node u in previous steps, it has a single chance to activate each inactive neighbour v with the successful probability $p(u, v)$. An activated node u remains active till the end of the diffusion process.
- The propagation process ends at step t if there is no new activated node in this step.

3.2 Problem definition

We adopt the Independent Cascade (IC) model to abstract the misinformation diffusion in a social network. In this problem, we denote $S \subseteq V$ is the *suspected set*, i.e, the set of nodes that is likely to be the source of misinformation. Each node $u \in S$ is a *source of misinformation* with probability $\rho(u) \geq 0$.

In IC model, we observe that the activations along edges are mutually independent. From the perspective of graph theory, the successful transmission from an user to its neighbors can be represented as an existence of the edges between them. Kempe et al. [20] show that IC model is equivalent to the reachability in a random graph g, called *live-edge graph* or *sample graph*. Accordingly, we can generate a sample graph g from original graph G, denoted by $g \sim G$, by selecting each edge $e = (u, v) \in E$, independently, with the probability $p(u, v)$, and no select edge (u, v) with the probability $1 - p(u, v)$. The probability that a realization g can be generated from G is:

$$\Pr[g \sim G] = \prod_{e \in E(g)} p(u, v) \prod_{e \in E \setminus E(g)} (1 - p(u, v)) \tag{1}$$

In above equation, $E(g)$ is the set node of g. The number of sample graphs is $2^{|E|}$. If we place a monitor on node v, it will detect misinformation from the nodes that are connected to it. It takes $d_g(u, v)$, the distance from u to v, hops to detect the misinformation from u. For a node $u \in S$, the probability that A can detect misinformation propagated from u is:

$$\mathbb{D}(A, u) = \sum_{g \sim G} \Pr[g \sim G] R(A, g, u) \tag{2}$$

where

$$R(A, g, u) = \begin{cases} 1, & \text{if } d_g(u, A) < \infty, \\ 0, & \text{Otherwise} \end{cases} \tag{3}$$

and $d_g(u, A) = \min_{v \in A} d_g(u, v)$. Since the probability that $u \in S$ is a source of misinformation node is $\rho(u)$, we define the *detection function* of A as follows:

$$\mathbb{D}(A) = \sum_{u \in S} \rho(u) \sum_{g \sim G} \Pr[g \sim G] R(A, g, u) \tag{4}$$

In this work, we study Minimum Budget for Misinformation Detection problem (MBD) defined as follows:

Definition 1 (MBD *problem*) Given a graph $G = (V, E)$ under IC model, a suspected set $S \subseteq V$ and each node $u \in S$ is a source of misinformation with probability $\rho(u) \geq 0$. Given a threshold for detection misinformation $\gamma > 0$, find the set of nodes $A \subseteq V$ with minimum-size to place monitors so that $\mathbb{D}(A) \geq \gamma$.

When all nodes have same probability to be the misinformation source, Zhang et. al [53] show that the detection function of a set of nodes A is equal to the influence spread of A on the reverse graph. Since calculating influence spread is #P-Hard [7], calculating detection function is also #P-Hard. Besides, we show the inapproximability of the problem by the following Theorem.

Theorem 1 (Inapproximability) *MBD cannot be approximated within a factor* $(1-\epsilon)\ln n$ *unless* $NP \in DTIME(n^{O\log\log n})$.

Proof We consider the decision version of MBD defined as follows: Given a graph $G=(V,E)$, a suspected set $S \subseteq V$, a threshold γ, and a positive number $k>0$. The problem asks whether or not the monitor set A of size k so that $\mathbb{D}(A) \geq \gamma$? □

We reduce MBD from the Set Cover problem defined as follows:

Definition 2 (*Set Cover (SC) problem*) Given a positive integer t, an universal set $\mathcal{U} = \{e_1, e_2, \ldots, e_M\}$ and a collection of subsets $\mathcal{S} = \{S_1, S_2, \ldots, S_N\}$. The Set Cover problem asks whether or not there are t subsets whose union is $\mathcal{U}$?

Reduction Given an instance $\mathcal{I} = (\mathcal{U}, \mathcal{S}, t)$ of SC problem, we construct an instance $\mathcal{I}' = (G, S, \gamma, k)$ as follows: For each element $e_i \in \mathcal{U}$, we create a node $u_i \in S$, and set $\rho(u) = 1$. For each subset $S_j \in \mathcal{S}$, we add a node v_j into S and add an edge (u_i, v_j) if $e_i \in S_j$ and set the probability $p(u_i, v_j) = 1$. For convenience, we denote sets $X = \{u_i, i = 1, \ldots, M\}$, and $Y = \{v_j, j = 1, \ldots, N\}$. Finally, we set $\gamma = M + k$ and $t = k$. We can see that the reduction can be done in polynomial-time respect to size of $\mathcal{I}$ and $\mathcal{I}'$.

Suppose that $\mathcal{I}$ has a solution $S' \subseteq \mathcal{S}$. By our reduction, in $\mathcal{I}'$ we choose a monitor set $A = \{v_j | S_j \in S'\}$. We have $\mathbb{D}(A) = M + t = \gamma$. This implies that A to be a solution of $\mathcal{I}'$. Conversely, suppose that $A \in V$ is the solution of $\mathcal{I}'$, i.e, $\mathbb{D}(A) \geq \gamma = M + k$ and $|A| = k$. Since each node $u_i \in X$ can only detect itself, A only contains some nodes in Y. From $|A| = k$, we imply that A can detect M node in X. Now, if we choose $S' = \{S_i | v_j \in A\}$, then S' is a solution of $\mathcal{I}$.

Suppose that there is an algorithm which can approximate MBD within a ratio of $(1-\epsilon)\ln n$ in polynomial time. By applying this algorithm and using our reduction, we can approximate SC within a ratio of $(1-\epsilon)\ln n$ in polynomial time. This contradicts to the fact that SC does not have a polynomial- time $(1-\epsilon)\ln n$ -approximation for any $\epsilon > 0$ unless NP $\in DTIME(n^{O\log\log n})$ [16]. □

4 Proposed algorithms

In this section, we propose three algorithms for MBD problem including Greedy, Sampling-based for Misinformation Detection (SMD) and Importance Sampling-based for Misinformation Detection (ISMD). Greedy provides an approximation ratio of $1+\ln(\gamma/\epsilon)$ but it cannot be applied to medium-networks even using the Monte-Carlo method to estimate the detection function because of its high complexity. SMD and ISMD are scalable algorithms with theoretical guarantees by developing a framework to select a final solution from multiple candidate solutions. The main difference between these algorithms is SMD estimates $\mathbb{D}(\cdot)$ by using the concept of detection sample while ISMD uses the concept of importance detection sample instead. Also, we show that ISMD takes lower complexity and uses fewer number of required samples than that of SMD.

4.1 Estimator of detection function

We introduce the concept of Detection Sampling (DS) set and use it to estimate $\mathbb{D}(\cdot)$.

Definition 3 (DS *set*) Given a graph $G = (V, E)$ under IC model, let $\rho(S) = \sum_{u \in S} \rho(u)$. A DS set R_j is generated from G by:

1. Picking a *source node* $u \in V$ with probability $\frac{\rho(u)}{\rho(S)}$.
2. Generating a sample graph g from G, and returning R_j as nodes which can be reached from u in g.

The meaning of the above definition is that each node in a DS set can detect misinformation spreading from u. Node u in the above definition is called the source of R_j, denoted by $src(R_j) = u$. We denote Ω is the probability space of all DS sets in which the probability of generating a DS set R_j having the source node u (denoted by $R_j(u)$) can be computed as follows:

$$\Pr[R_j(u) \sim \Omega] = \frac{\rho(u)}{\rho(S)} \cdot \sum_{g \sim G: R(R_j, g, u) = 1} \Pr[g \sim G] \tag{5}$$

If we generate multiple DS sets, the nodes that can detect misinformation from many other nodes will likely appear frequently in the DS sets. Basically, the role of DS is similar to the Reachable Reverse (RR) set in estimating influence spread function [3,36,47–49]. We define a random variable $X_j(A)$ as follows:

$$X_j(A) = \begin{cases} 1, & \text{If } R_j \cap A \neq \emptyset \\ 0, & \text{Otherwise} \end{cases} \tag{6}$$

Similar to Lemma 2 in [3], Lemma 1 shows that we can use the value of $X_j(A)$ to estimate function $\mathbb{D}(A)$.

Lemma 1 *For any set of nodes $A \subseteq V$, let $\rho(S) = \sum_{u \in S} \rho(u)$ we have:*

$$\mathbb{D}(A) = \rho(S) \cdot \mathbb{E}[X_j(A)] \tag{7}$$

Proof Since the source node u is chosen with probability $\frac{\rho(u)}{\rho(S)}$, we have:

$$\mathbb{D}(A) = \sum_{u \in V} \rho(u) \sum_{g \sim G} \Pr[g \sim G] R(A, g, u) \tag{8}$$

$$= \sum_{g \sim G} \sum_{u \in V} \rho(u) R(A, g, u) \Pr[g \sim G] \tag{9}$$

$$= \sum_{g \sim G} \rho(S) \sum_{u \in V} R(A, g, u) \Pr[g \sim G] \frac{\rho(u)}{\rho(S)} \tag{10}$$

$$= \rho(S) \sum_{g \sim G} \sum_{u \in V} \mathsf{Cov}(R_j^g(u), A) \Pr[g \sim G] \Pr[u \text{ is the source node}] \tag{11}$$

$$= \rho(S) \cdot \mathbb{E}[X_j(A)] \tag{12}$$

where $R_j^g(u)$ is a DS corresponding to the sample graph g and the source node u. □

4.2 Greedy algorithm

We introduce Greedy algorithm that provides an approximation ratio of $1 + \ln(\gamma/\epsilon)$ based on the *submodular* and *monotone* properties of the $\mathbb{D}(\cdot)$ function, i.e, for $A \subseteq T \subseteq V$, $v \notin T$ $\mathbb{D}(A + \{v\}) - \mathbb{D}(A) \geq \mathbb{D}(T + \{v\}) - \mathbb{D}(T)$.

Lemma 2 *The function $\mathbb{D}(A)$ is monotone and submodular.*

Proof Rewrite Eq. (11), we have:

$$\mathbb{D}(A) = \sum_{g \sim G} \sum_{u \in V} \Pr[g \sim G] \rho(u) \mathsf{Cov}(R_j^g(u), A) \tag{13}$$

the above equation shows that the detection function $\mathbb{D}(A)$ is equivalent to the weighted coverage function of a set cover system in which: every R_j is an element in the set of all DS set and each node in V is a subset and V is a collection of subsets. Each node v covers set R_j if v belongs to R_j. The value of $\Pr[g \sim G]\rho(u)$ is the weight of an element $R_j^g(u)$. Since the weighted coverage function is monotone and submodular, it has the same properties with $\mathbb{D}(A)$. □

Lemma 2 help us design an $(1+\ln \frac{\gamma}{\epsilon})$-approximation algorithm by applying Greedy algorithm in [18] (Algorithm 1), where $\epsilon \in (0, \gamma)$ is an input.

At the beginning, this algorithm initiates a solution A as an empty set. The main phase of the algorithm operates in multiple iterators (line 2-5). In each iterator, it simply chooses a node u that provides the largest *incremental detection function*, defined as follows:

$$\delta(A, u) = \min(\mathbb{D}(A \cup \{u\}), \gamma) - \mathbb{D}(A) \tag{14}$$

until $\mathbb{D}(A) \geq \gamma - \epsilon$. However, we can not implement Greedy even for small networks because calculating the detection function is #P-hard. To address this challenge, we can use the Monte Carlo simulation method to estimate the detection function. Let R be the maximum time needed to estimate the $\mathbb{D}(\cdot)$ by using Mote-Carlo simulation method, Greedy takes $O(Rnk)$ time complexity, where k is the number of iterators in algorithm.

4.3 Sampling-based for misinformation detection algorithm

This algorithm combines two novel techniques: (1) generating a collection of DS sets that is enough to estimate the detection function by applying martingale theory and (2) a new framework for generating candidate solutions and checking their quality to select the final solution. Denote $\mathsf{Cov}_{\mathcal{R}}(A) = \sum_{R_j \in \mathcal{R}} \min\{1, |A \cap R_j|\}$ as the number

Algorithm 1: Greedy algorithm

Input: A graph $G = (V, E)$, a suspected set $S \subseteq V$, a threshold γ, $\epsilon \in (0, \gamma)$
Output: A set node A

1. $A \leftarrow \emptyset$
2. **while** $\mathbb{D}(A) < \gamma - \epsilon$ **do**
3. $\quad u \leftarrow \arg\max_{v \in V \setminus S}(\min(\mathbb{D}(A \cup \{v\}), \gamma) - \mathbb{D}(A))$
4. $\quad A \leftarrow A \cup \{u\}$
5. **end**
6. **return** A;

of DS sets in $\mathcal{R}$ covered by A. From Lemma 1 we obtain an estimation of $\mathbb{D}(A)$ from $\mathcal{R}$ as follows:

$$\hat{\mathbb{D}}(A) = \frac{\rho(S)}{|\mathcal{R}|}\mathsf{Cov}_{\mathcal{R}}(A) \tag{15}$$

Since $\mathsf{Cov}_{\mathcal{R}}(\cdot)$ is monotone and submodular, $\hat{\mathbb{D}}(\cdot)$ is also monotone and submodular.

Detection sampling algorithm We first devise an algorithm for generating a DS which is inspired by the Breath-First-Search (BFS) algorithm, formally described below as Algorithm 2. It first selects a source node u with probability $\frac{\rho(u)}{\rho(S)}$ (line 1), then uses a queue Q to store the visited nodes and initiates a DS set $R_j = \{u\}$. The rest of this algorithm operates in several iterators. At each iterator, it picks a node u from Q and adds u into R_j, then selects each neighbor node v (not belong to Q) with probability $p(u, v)$ according to the live-edge model (line 8). If v is selected, it is put into Q. Otherwise, the algorithm moves to next iterator. This process repeats until Q becomes an empty set.

Algorithm 2: Detection Sampling algorithm

Input: A graph $G = (V, E)$, a suspected set $S \subseteq V$
Output: A DS set R_j

1. Select a node $u \in V$ with probability $\Pr[u] = \frac{\rho(u)}{\rho(S)}$
2. Queue $Q \leftarrow \{u\}$;
3. **while** Q *is not empty* **do**
4. $\quad u \leftarrow Q.pop()$
5. $\quad R_j \leftarrow R_j \cup \{u\}$
6. $\quad$ **foreach** $v \in N_{out}(u) \setminus R_j$ **do**
7. $\quad\quad$ **if** $v \notin Q$ **then**
8. $\quad\quad\quad$ Select v with probability $p(u, v)$
9. $\quad\quad\quad$ **if** *(v is selected)* **then**
10. $\quad\quad\quad\quad Q.push(v)$
11. $\quad\quad\quad$ **end**
12. $\quad\quad$ **end**
13. $\quad$ **end**
14. **end**
15. **return** R_j;

4.3.1 Description of SMD algorithm

Algorithm 3: Sampling-based Misinformation Detection (SMD) algorithm

Input: A graph $G = (V, E)$, a suspested set $S \subseteq V$, a threshold $\gamma > 0$, $\epsilon, \delta \in (0, 1)$
Output: A set node A

1. $N \leftarrow \frac{(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(n/\delta)$
2. Generate set $\mathcal{R}$ containing N DS sets by Alg. 2
3. $A \leftarrow \emptyset$
4. **while** *True* **do**
5. $\quad u \leftarrow \arg\max_{v \in V \setminus A} \left(\hat{\mathbb{D}}(A \cup v) - \hat{\mathbb{D}}(A)\right)$; `// $\hat{\mathbb{D}}(A)$ is calculated by Eq. (15)`
6. $\quad A \leftarrow A \cup \{u\}$
7. $\quad$ **if** $\hat{\mathbb{D}}(A) \geq (\gamma - \epsilon\gamma) - \epsilon$ **then**
8. $\quad\quad$ **return** A
9. $\quad$ **else**
10. $\quad\quad i \leftarrow |A| + 1$
11. $\quad\quad N_i \leftarrow \frac{(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(\binom{n}{i}/\delta)$
12. $\quad\quad$ **if** $N < N_i$ **then**
13. $\quad\quad\quad$ Generate more $N_i - N$ DS sets and add them into $\mathcal{R}$
14. $\quad\quad\quad N \leftarrow N_i$
15. $\quad\quad\quad A \leftarrow \emptyset$
16. $\quad\quad$ **end**
17. $\quad$ **end**
18. **end**
19. **return** A;

This algorithm first generates a set $\mathcal{R}$ containing $N = \frac{(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(n/\delta)$ DS sets which ensures (δ, ϵ)-approximation for optimal solution A^* (Lemma 4), i.e,

$$\Pr[(1+\epsilon)\mathbb{D}(A^*) \geq \hat{\mathbb{D}}(A^*) \geq (1-\epsilon)\mathbb{D}(A^*)] \geq 1 - \delta \tag{16}$$

The algorithm initiates an empty candidate solution A and its main phase happens in several iterators (line 4-18) to select the final solution from multiple candidate solutions.

Firstly, we observe that the candidate solutions may have different sizes and we do not know the size of the final solution. Therefore, in each iterator i, we maintain a set $\mathcal{R}$ with the size at least:

$$N_i(\delta, \epsilon) = \frac{(2+\frac{2}{3}\epsilon)n}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(\binom{n}{i}/\delta)$$

which guarantees the bi-criterion approximation (Theorem 2) for the candidate solution A with size $|A| = i$. The algorithm then selects node u, which provides the largest incremental of estimation of detection function $\hat{\delta}(A, v) = \hat{\mathbb{D}}(A \cup v) - \hat{\mathbb{D}}(A)$ into the candidate solution A (line 5).

The algorithm then checks the quality of the candidate solution A in line 7. If the set A satisfies $\hat{\mathbb{D}}(A) \geq \gamma - \epsilon\gamma - \epsilon$ then the algorithm returns A. If not, it checks whether the current number of samples is enough or not for the next iterator (the size of candidate solution increasing by 1) (line 12). If yes, it moves to next iterators. If not, it generates more $N_i - N$ DS sets, adds them into $\mathcal{R}$ (line 13) and resets current solution A, i.e, it sets A as an empty set (line 15). The algorithm moves to next iterator and constructs an another candidate solution from an empty set. The algorithm terminates only when it meets the condition $\hat{\mathbb{D}}(A) \geq \gamma - \epsilon\gamma - \epsilon$.

4.3.2 Theoretical analysis

We now analyze the approximation guarantee of SMD using the martingale theory [11] which is used for studying information propagation problems [36–38,47,49].

Definition 4 (*Martingale*) A sequence of random variable $T_1, T_2, T_3, \ldots, T_l$ is a martingale, if only if $\mathbb{E}[T_i] \leq +\infty$ and $\mathbb{E}[T_i|T_1, T_2, T_3, \ldots, T_{i-1}] = T_{i-1}$ for any $i = 2 \ldots l$.

The following concentration inequality [11] for martingales that have similar flavor to the Chernoff bounds.

Lemma 3 ([11], Theorem 6.1) *If $T_1, T_2, \ldots, T_l$ be a form of martingale satisfying*

1) $|T_1| \leq a$, $|T_j - T_{j-1}| \leq a$, for $1 \leq j \leq l$
2) $\mathsf{Var}[T_j|T_1, T_2, \ldots, T_{j-1}] \leq \sigma_i^2$, for $1 \leq j \leq l$

where $\mathsf{Var}[\cdot]$ denotes the variance of a random variable. Then, for any λ, we have:

$$\Pr[T_i - \mathbb{E}[T_i] \geq \lambda] \geq \exp\left(-\frac{\lambda^2}{\frac{2}{3}a\lambda + 2\sum_{i=1}^{l}\sigma_i^2}\right) \tag{17}$$

Given a collection of DS sets $\mathcal{R}$, we consider a sequence of the random variables $\{X_j(A)\}$, $j = 1, \ldots, |\mathcal{R}|$. We observe that $X_j(A) \in [0, 1]$, let a random variable $M_i = \sum_{j=1}^{i}(X_j(A) - \mu_X), \forall i \geq 1$, where $\mu_X = \mathbb{E}[X_j]$. For a sequence of random variables $M_1, M_2, \ldots, M_{|\mathcal{R}|}$, we have

$$\mathbb{E}[M_i|M_1, \ldots, M_{i-1}] = \mathbb{E}[M_{i-1}] + \mathbb{E}[X_i(A) - \mu_X] = \mathbb{E}[M_{i-1}], \forall i = 2, \ldots, |\mathcal{R}|$$

Hence, $M_1, M_2, \ldots, M_{|\mathcal{R}|}$ be a form of martingale [11]. Apply Lemma 3 with $a = 1$, $\mathsf{Var}[M_j|M_1, M_2, \ldots, M_{j-1}] = \mathsf{Var}[X_j(A) - \mu_X] = \mathsf{Var}[X_j(A)] \leq 1$, $l = |\mathcal{R}|$ and $\lambda = \epsilon|\mathcal{R}|\mu_X$ we have

$$\Pr[\sum_{j=1}^{|\mathcal{R}|} X_j(A) \geq (1+\epsilon)\mu_X|\mathcal{R}|] \leq \exp\left(\frac{-\epsilon^2|\mathcal{R}|\mu_X}{2 + \frac{2}{3}\epsilon}\right) \tag{18}$$

Similarly, $-M_1, \ldots, -M_i, \ldots, -M_{|\mathcal{R}|}$ also form a martingale and applying Lemma 3 will gives the following probabilistic inequality.

$$\Pr[\sum_{j=1}^{|\mathcal{R}|} X_j(A) \leq (1-\epsilon)\mu_X|\mathcal{R}|] \leq \exp\left(\frac{-\epsilon^2|\mathcal{R}|\mu_X}{2}\right) \tag{19}$$

By applying two inequalities above, we obtain the following Lemma indicating a importance property of the optimal solution.

Lemma 4 *Given* $\epsilon, \delta \in (0, 1)$. *If* $|\mathcal{R}| \geq \frac{(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln\frac{1}{\delta}$, *we have*

$$\Pr[\hat{\mathbb{D}}(A^*) \geq \gamma - \epsilon\gamma] \geq 1 - \delta \tag{20}$$

where $\hat{\mathbb{D}}(A^*)$ *is calculated by Eq.* (15) *and* A^* *is an optimal solution.*

Proof Denote $\mu = \mathbb{D}(A^*)/\rho(S)$, $\hat{\mu} = \hat{\mathbb{D}}(A^*)/\rho(S)$, apply (19) we have

$$\begin{aligned}
\Pr[\hat{\mathbb{D}}(A^*) \leq \gamma - \epsilon\gamma] &\leq \Pr[\hat{\mathbb{D}}(A^*) \leq (1-\epsilon)\mathbb{D}(A^*)] && (21)\\
&= \Pr[\hat{\mu} \leq (1-\epsilon)\mu] \leq \exp\left(\frac{-\epsilon^2|\mathcal{R}|\mu}{2}\right) && (22)\\
&\leq \exp\left(\frac{-\epsilon^2|\mathcal{R}|\hat{\mu}}{2(1-\epsilon)}\right) && (23)\\
&\leq \exp\left(-\frac{(2+\frac{2}{3}\epsilon)\hat{\mathbb{D}}(A^*)}{2(1-\epsilon)(\gamma-\epsilon\gamma)} \ln\frac{1}{\delta}\right) \leq \delta && (24)
\end{aligned}$$

This completes the proof □

Theorem 2 *Given* $\epsilon, \delta \in (0, 1)$, *the Algorithm* 3 *returns a solution* A *with*

a) $\Pr[|A| \leq 1 + |A^*| \cdot \ln\frac{\gamma-\gamma\epsilon}{\epsilon}] \geq 1 - \frac{\delta}{n}$.
b) $\Pr\left(\mathbb{D}(A) \geq \gamma \cdot \frac{1-\epsilon}{1+\epsilon} - \epsilon\right) \geq 1 - \delta$.

Proof We consider the case when while-loop (line 4-18) terminates. Assume that the algorithm returns a solution $A = \{a_1, a_2, \ldots, a_p\}$, denote $A_i = \{a_1, a_2, \ldots, a_i\}$, $i \leq p$, we have the number of samples $|\mathcal{R}| = \frac{(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(\binom{n}{i_{max}}/\delta)$, where

$$i_{max} = \arg\max_{i:1\ldots p} \frac{(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln\frac{\binom{n}{i}}{\delta} \tag{25}$$

Proof a) Let $B = \{b_1, b_2, \ldots, b_l\}$ be a set of minimum size satisfying $\hat{\mathbb{D}}(B) \geq \gamma - \epsilon\gamma$, we have:

$$\gamma - \epsilon\gamma - \hat{\mathbb{D}}(A_i) \leq \hat{\mathbb{D}}(A_i \cup B) - \hat{\mathbb{D}}(A_i)$$

$$
\begin{aligned}
&= \sum_{j=1}^{l} \left(\hat{\mathbb{D}}(A_i \cup \{b_1, b_2, \ldots, b_j\}) - \hat{\mathbb{D}}(A_i \cup \{b_1, b_2, \ldots, b_{j-1}\}) \right) \\
&\leq \sum_{j=1}^{l} (\hat{\mathbb{D}}(A_i \cup \{b_j\}) - \hat{\mathbb{D}}(A_i)) \quad \text{(Since } \hat{\mathbb{D}}(\cdot) \text{ is submodular)} \\
&\leq l \cdot (\hat{\mathbb{D}}(A_i) - \hat{\mathbb{D}}(A_{i-1}))
\end{aligned}
$$

Therefore,

$$
\gamma - \epsilon\gamma - \hat{\mathbb{D}}(A_i) \leq (1 - \frac{1}{l}) \left(\gamma - \gamma\epsilon - \hat{\mathbb{D}}(A_{i-1}) \right) \tag{26}
$$

$$
= \left(1 - \frac{1}{l}\right)^i (\gamma - \gamma\epsilon) \leq e^{-i/l}(\gamma - \gamma\epsilon) \tag{27}
$$

Because the candidate solution A meets condition in line 7, and by the definition of A_i, we have $\hat{\mathbb{D}}(A_p) \geq \gamma - \epsilon\gamma - \epsilon$ and $\hat{\mathbb{D}}(A_{p-1}) < \gamma - \epsilon\gamma - \epsilon$. Combine with (27), we have:

$$
(\gamma - \gamma\epsilon)e^{\frac{-p-1}{l}} \geq (\gamma - \gamma\epsilon) - \hat{\mathbb{D}}(A_{p-1}) \geq (\gamma - \epsilon\gamma) - (\gamma - \epsilon\gamma - \epsilon) = \epsilon \tag{28}
$$

$$
\implies p \leq 1 + l \cdot \ln \frac{\gamma - \gamma\epsilon}{\epsilon} \tag{29}
$$

From Lemma 4, we have $\Pr[\hat{\mathbb{D}}(A^*) \geq \gamma - \epsilon\gamma] \geq 1 - \delta/\binom{n}{i_{max}}$. Due to the definition of B, the following event happens with the probability at least $1 - \delta/\binom{n}{i_{max}}$

$$
|A| \leq 1 + |B| \ln \frac{\gamma - \gamma\epsilon}{\epsilon} \leq 1 + |A^*| \ln \frac{\gamma - \gamma\epsilon}{\epsilon} \tag{30}
$$

Hence, $\Pr[|A| \leq 1 + |A^*| \cdot \ln \frac{\gamma - \gamma\epsilon}{\epsilon}] \geq 1 - \delta/\binom{n}{i_{max}} \geq 1 - \delta/n$.

Proof b) Since $\hat{\mathbb{D}}(A) \geq \gamma - \epsilon\gamma - \epsilon$ where A is a solution returned by Algorithm 3. Therefore,

$$
\begin{aligned}
\Pr\left(\mathbb{D}(A) \leq \gamma \frac{1-\epsilon}{1+\epsilon} - \epsilon\right) &\leq \Pr\left(\mathbb{D}(A) \leq \frac{\gamma - \gamma\epsilon - \epsilon}{1+\epsilon}\right) \leq \Pr\left(\mathbb{D}(A) \leq \frac{\hat{\mathbb{D}}(A)}{1+\epsilon}\right) \\
&= \Pr[\hat{\mathbb{D}}(A) \geq (1+\epsilon)\mathbb{D}(A)] = \Pr[\hat{\mu} \geq (1+\epsilon)\mu] \\
&\leq \exp\left(\frac{-\epsilon^2|\mathcal{R}|\mu}{2 + \frac{2}{3}\epsilon}\right) \leq \exp\left(\frac{-\epsilon^2|\mathcal{R}|\hat{\mu}}{(2 + \frac{2}{3}\epsilon)(1+\epsilon)}\right) \\
&= \exp\left(-\frac{\ln \frac{\binom{n}{i_{max}}}{\delta}}{(1+\epsilon)}\right) \leq \frac{\delta}{\binom{n}{i_{max}}} \leq \frac{\delta}{\binom{n}{p}}
\end{aligned}
$$

Since $|A| = p$, there are at most $\binom{n}{p}$ possible solutions. Therefore,

$$\Pr\left(\exists A, \mathbb{D}(A) \leq \gamma \cdot \frac{1-\epsilon}{1+\epsilon} - \epsilon\right) \leq \delta \tag{31}$$

Hence,

$$\Pr\left(\forall A, \mathbb{D}(A) \geq \gamma \cdot \frac{1-\epsilon}{1+\epsilon} - \epsilon\right) \geq 1 - \delta \tag{32}$$

The proof is completed □

Complexity of SMD ***algorithm*** The number of required samples in the worst-case $\frac{(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(\binom{n}{i_{max}}/\delta) = O(\rho(S)\ln(\binom{n}{i_{max}}/\delta)\epsilon^{-2})$. Denote M is the expected running time for generating one sample, the time complexity of Algorithm 3 is

$$O\left(i_{max}\rho(S)\ln(\binom{n}{i_{max}}/\delta)\epsilon^{-2}M\right)$$

4.4 Importance sampling-based misinformation detection algorithm

We next introduce the ISMD algorithm, an improved version of SMD, which provides the same approximation guarantee with SMD but requires fewer samples than SMD. The main idea of this algorithm is that we propose an importance detection sample (IDS) concept to estimate $\mathbb{D}(\cdot)$ function instead of DS.

4.4.1 Importance detection sampling

We observe that DS sets contain only one node contributing insignificantly in calculating the detection function. Therefore, we only consider the generation of DS sets that contain more than one node, called IDS sets. We show that the detection function can be estimated through the IDS sets (Lemma 5).

We now describe how to generate an IDS. For a source node u, Ω_u denotes the set of all DS sets that have a source u. We divide Ω_u into two components:

- Trivial samples: the set contains only a source node u, called Ω_u^0.
- Importance samples: $\Omega_u^i = \Omega_u \setminus \Omega_u^0$.

For a source node u, let E_0 be the event that none of nodes in $N_{out}(u)$ is activated by u, we have:

$$\Pr[E_0] = \prod_{v \in N_{out}(u)} (1 - p(u, v)) \tag{33}$$

The probability that at least a node in $N_{out}(u)$ is influenced by u is equal to the probability of generating an importance detection sample from u:

$$\varphi(u) = 1 - \Pr[E_0] = 1 - \prod_{u \in N_{out}(u)} (1 - p(u, v)) \tag{34}$$

To generate a IDS set, we construct Ω_u^i according to following analysis. Assume that $N_{out}(u) = \{v_1, v_2, \ldots, v_{l(u)}\}$ with $|N_{out}(v)| = l(u)$, suppose E_i as the event that v_i is the first node in $N_{out}(u)$ which is influenced by u, we have:

$$\Pr[E_i] = p(u, v_i) \cdot \prod_{j=1}^{i-1} (1 - p(u, v_i)) \tag{35}$$

By definition of E_i, we have:

$$\sum_{i=1}^{l(u)} \Pr[E_i]/\varphi(u) = 1 \quad \text{end} \quad E_i \cap E_j = \emptyset \tag{36}$$

Ω_n denotes the probability spaces of all IDS samples. The probability that an IDS set R_j with the source node u generated from Ω_n is

$$\Pr[R_j(u) \sim \Omega_n] = \frac{1}{\varphi(u)} \Pr[R_j(u) \sim \Omega] \tag{37}$$

The probability that a node u is a source node of an IDS R_j in Ω is $\frac{\rho(u)}{\rho(S)}\varphi(u)$. By normalizing factor to fulfill a probability distribution of all IDS samples, the probability that u is the source node of a IDS R_j in Ω_n is:

$$\Pr[src(R_j) = u] = \frac{\rho(u)\varphi(u)}{\sum_{v \in V} \rho(v)\varphi(v)} = \frac{\rho(u)\varphi(u)}{\Phi} \tag{38}$$

Where $\Phi = \sum_{v \in V} \rho(v)\varphi(v)$. For any IDS set R_j, we have:

$$\Pr[R_j \sim \Omega] = \sum_{u \in V} \Pr[u \text{ is source of } R_j \text{ in } \Omega] \Pr[R_j(u) \sim \Omega] \tag{39}$$

$$= \sum_{u \in V} \frac{\rho(u)}{\rho(S)} \cdot \varphi(u) \Pr[R_j(u) \sim \Omega_n] \tag{40}$$

$$= \frac{\Phi}{\rho(S)} \cdot \sum_{u \in V} \frac{\rho(u)\varphi(u)}{\Phi} \Pr[R_j(u) \sim \Omega_n] \tag{41}$$

$$= \frac{\Phi}{\rho(S)} \cdot \sum_{u \in V} \Pr[u \text{ is source of } R_j \text{ in } \Omega_n] \Pr[R_j(u) \sim \Omega_n] \tag{42}$$

$$= \frac{\Phi}{\rho(S)} \cdot \Pr[R_j \sim \Omega_n] \tag{43}$$

We define two random variables $Z_j(A)$ and $Y_j(A)$ as follows:

$$Z_j(A) = \begin{cases} 1, & \text{If } R_j \cap A \neq \emptyset \\ 0, & \text{Otherwise} \end{cases} \tag{44}$$

And,

$$Y_j(A) = \frac{\Phi \cdot Z_j(A) + \sum_{v \in A}(1 - \varphi(v))\rho(v)}{\rho(S)} \tag{45}$$

We have $Y_j(A) \in [Y_{min}, Y_{max}]$, with $Y_{min} = \frac{\sum_{v \in A}(1-\varphi(v))\rho(v)}{\rho(S)}$, $Y_{max} = \frac{\Phi + \sum_{v \in A}(1-\varphi(v))\rho(v)}{\rho(S)}$, we have following Lemma:

Lemma 5 *For any set of nodes $A \subseteq V$, we have:*

$$\mathbb{D}(A) = \Phi \cdot \mathbb{E}[Z_j(A)] + \sum_{v \in A}(1 - \varphi(v))\rho(v) = \rho(S) \cdot \mathbb{E}[Y_j(A)] \tag{46}$$

Proof From Lemma 1, we have

$$\mathbb{D}(A) = \rho(S) \cdot \sum_{R_j \in \Omega} \Pr[R_j \sim \Omega] X_j(A) \tag{47}$$

$$= \rho(S) \cdot \left(\sum_{R_j \in \Omega_0} \Pr[R_j \sim \Omega] X_j(A) + \sum_{R_j \in \Omega_n} \Pr[R_j \sim \Omega] X_j(A) \right) \tag{48}$$

Since each $R_j \in \Omega_0$ contains only a source node, we have:

$$\Pr[R_j \sim \Omega] = \frac{\rho(u)}{\rho(S)}(1 - \varphi(u))$$

where $u = src(R_j)$. Put it into (48), we have:

$$\begin{aligned} \mathbb{D}(A) &= \rho(S) \sum_{u \in A} \frac{\rho(u)}{\rho(S)}(1 - \varphi(u)) + \rho(S) \sum_{R_j \in \Omega_n} \Pr[R_j \sim \Omega] \mathsf{Cov}(A, R_j) \\ &= \sum_{u \in A} \rho(u)(1 - \varphi(u)) + \rho(S) \sum_{R_j \in \Omega_n} \frac{\Phi}{\rho(S)} \Pr[R_j \sim \Omega_n] \mathsf{Cov}(A, R_j) \\ &= \sum_{u \in A} \rho(u)(1 - \varphi(u)) + \Phi \sum_{R_j \in \Omega_n} \Pr[R_j \sim \Omega_n] Z_j(A) \end{aligned}$$

$$= \sum_{u \in A} \rho(u)(1 - \varphi(u)) + \Phi \mathbb{E}[Z_j(A)]$$
$$= \rho(S) \cdot \mathbb{E}[Y_j(A)] \text{ (Due to the definition of } Y_j(A))$$

which competes the proof. □

From Lemma 5, we have another estimation of $\mathbb{D}(A)$ by utilizing a set of IDS $\mathcal{R}$:

$$\hat{\mathbb{D}}(A) = \frac{\Phi}{|\mathcal{R}|} \sum_{R_j \in \mathcal{R}} \mathsf{Cov}(A, R_j) + \sum_{v \in A} (1 - \varphi(v))\rho(v) \tag{49}$$

From the above analysis, we propose an Importance Detection Sampling algorithm to generate an IDS set by modifying Algorithm 2. The details of this algorithm are described in Algorithm 4.

The algorithm first selects a source of IDS with the probability according to e.q (38) (line 1). Then, it calculates probabilities $\Pr[E_i], i = 1 \ldots, l(u)$ and selects a first out-neighbour u_i in $N_{out}(u)$ with probability $\Pr[E_i]/\varphi(u)$ (line 3). This guarantees that at least one of the out-neighbors of u will be selected. Similar to Alg. 2, this algorithm also uses a queue Q to store the visited nodes and initiates an IDS $R_j = \{u\}$. At this time, $Q = R_j = \{u, v_i\}$ (line 4-5). The nodes $v_1, v_2, \ldots, v_{j-1}$ are ignored and nodes $v_{i+1}, \ldots, v_l$ are then selected independently with probability $p(u, v_j)$, $j = i + 1 \ldots l$ (line 7) and added into Q and R_j (line 9). The rest of this algorithm is similar to the while-loop (line 3-14) in Alg. 2 because of the similarity between IDS and DS from this step.

4.4.2 Description of ISMD algorithm

The details of ISMD is presented in Algorithm 5. This algorithm works similarly to SMD algorithm, the main difference between these two algorithms lies in the following two factors. Firstly, ISMD generates IDS sets instead of DS (line 2) and uses them for estimating the detection function by Eq. (49). Secondly, the number of required samples of ISMD in each iterator is lower than that of SMD. Specifically, ISMD needs $\frac{q(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(\binom{n}{i}/\delta)$, $(q < 1)$ samples, which is fewer than that of SMD a factor of q, $(q < 1)$.

4.4.3 Theoretical analysis

We show that ISMD has the same approximation guarantee with SMD but ISMD needs fewer samples than SMD. From that on, we also point out that the complexity of ISMD is less than that of SMD. Firstly, by applying Lemma 3, we have following Lemma

Algorithm 4: Importance Detection Sampling algorithm

Data: Graph $G = (V, E)$, suspected set $S \subseteq V$
Result: an IDS set R_j

Select a node $u \in V$ with probability $\Pr[u] = \frac{\varphi(u)\rho(u)}{\Phi}$
Calulate $\Pr[E_i], i = 1 \ldots l(u)$ by Eq. (35).
Select one node $v_i \in N_{out}(u)$ with probability $\frac{\Pr[E_i]}{\varphi(u)}$
$R_j \leftarrow \{u, v_i\}$;
Queue $Q \leftarrow \{v_i\}$;
for $j = i + 1$ *to* l **do**
    Select v_j with propability $p(u, v_j)$;
    if *(v_j is selected)* **then**
        $Q.push(v_j)$, $R_j \leftarrow R_j \cup \{v_j\}$
    end
end
while Q *is not empty* **do**
    $u \leftarrow Q.pop()$
    foreach $v \in N_{out}(u) \setminus R_j$ **do**
        if $v \notin Q$ **then**
            Select v with probability $p(u, v)$
            if *(v is selected)* **then**
                $Q.push(v)$
                $R_j \leftarrow R_j \cup \{v\}$
            end
        end
    end
end
return R_j;

Lemma 6 *For any $T = |\mathcal{R}| > 0$, $\lambda > 0$, μ is the mean of $Y_j(A)$, and an estimation of μ is $\hat{\mu} = \frac{\sum_{i=1}^{T} Y_j(A)}{T}$. Let $q = Y_{max} - Y_{min} = \frac{\Phi}{\rho(S)}$, we have:*

$$\Pr\left[\sum_{j=1}^{T} Y_j(A) - T \cdot \mu \geq \lambda\right] \leq \exp\left(-\frac{\lambda^2}{\frac{2}{3}q\lambda + 2q\mu T}\right) \tag{50}$$

$$\Pr\left[\sum_{j=1}^{T} Y_j(A) - T \cdot \mu \geq -\lambda\right] \leq \exp\left(-\frac{\lambda^2}{2q\mu T}\right) \tag{51}$$

Proof For any set $A \subseteq V$, since $Y_j(A) \in [Y_{min}, Y_{max}]$ we have

$$\mathsf{Var}[Y_j(A)] \leq (\mu - Y_{min})(Y_{max} - \mu) \leq (Y_{max} - Y_{min})\mu = q \cdot \mu \tag{52}$$

Choose randomly variable $M_i = \sum_{j=1}^{i}(Y_j(A) - \mu), \forall i \geq 1$, where $\mu = \mathbb{E}[Y_j]$. We can easily show that $M_1, M_2, \ldots$ is a form of martingale [11]. Applying Lemma 3, with $a = q$, $\mathsf{Var}[M_j|M_1, M_2, \ldots, M_{j-1}] = \mathsf{Var}[Y_j(A) - \mu] = \mathsf{Var}[Y_j(A)] \leq q$,

Algorithm 5: Importance Sampling-Based for Misinformation Detection (ISMD) algorithm

Input: A graph $G = (V, E)$, a suspested set $S \subseteq V$, a threshold $\gamma > 0, \epsilon, \delta \in (0, 1)$
Output: A set node A

$N \leftarrow \frac{q(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(n/\delta)$
Generate set $\mathcal{R}$ containing N IDS set
$A \leftarrow \emptyset$
while *True* **do**
  $u \leftarrow \arg\max_{v \in V \setminus A} \left(\hat{\mathbb{D}}(A \cup v) - \hat{\mathbb{D}}(A)\right)$; `// $\hat{\mathbb{D}}(A)$ is calculated by Eq. (49)`
  $A \leftarrow A \cup \{u\}$
  if $\hat{\mathbb{D}}(A) \geq \gamma - \epsilon\gamma - \epsilon$ **then**
    return A
  else
  end
  $i \leftarrow |A| + 1$
  $N_i \leftarrow \frac{q(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(\binom{n}{i}/\delta)$
  if $N < N_i$ **then**
    Generate more $N_i - N$ IDS sets and add them into $\mathcal{R}$
    $N \leftarrow N_i$
    $A \leftarrow \emptyset$
  end
end
return A;

$T = l$ and $\lambda = \epsilon T \mu_X$ we have

$$\mathsf{Var}[M_1] + \sum_{j=2}^{i} \mathsf{Var}[M_j | M_1, M_2, \ldots, M_{j-1}] = \sum_{j=1}^{T} \mathsf{Var}[T_j(A)] \leq q\mu T \tag{53}$$

Applying Lemma 3 with $a = q$ and $b = Tq\mu$, and put back into (17) we obtain (50). Similarly, $-M_1, \ldots, -M_i, \ldots$ also form a martingale and by applying (17), we obtain (51). □

Lemma 7 *Given* $\epsilon, \delta \in (0, 1)$. *If* $|\mathcal{R}| \geq \frac{q(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln \frac{1}{\delta}$, *we have* $\Pr[\hat{\mathbb{D}}(A^*) \geq \gamma - \epsilon\gamma] \geq 1 - \delta$ *where* $\hat{\mathbb{D}}(A)$ *is calculated by* (49).

Proof Applying Lemma 6, with $\lambda = \epsilon T \mu$, $q = \frac{\Phi}{\rho(S)}$ we have

$$\Pr[\hat{\mathbb{D}}(A^*) \leq \gamma - \epsilon\gamma] \leq \Pr[\hat{\mathbb{D}}(A^*) \leq (1-\epsilon)\mathbb{D}(A^*)] \tag{54}$$

$$= \Pr[\hat{\mu} \leq (1-\epsilon)\mu] \leq \exp\left(\frac{-\epsilon^2|\mathcal{R}|\mu}{2q}\right) \tag{55}$$

$$\leq \exp\left(-\frac{(2+\frac{2}{3}\epsilon)\hat{\mathbb{D}}(A^*)}{2(1-\epsilon)(\gamma-\epsilon\gamma)} \ln \frac{1}{\delta}\right) \leq \delta \tag{56}$$

Table 2 Datasets

Dataset	Nodes	Edges	Type	Avg. degree
Email-Eu-Core [24,51]	1005	25,571	Directed	25.44
Wiki-Vote [26,27]	7115	103,689	Directed	14.57
CA-HepPh [24]	12,008	118,521	Undirected	9.87
CA-AstroPh [24]	18,722	198,110	Undirected	10.58
Email-Eu-All [24]	265,214	420,045	Directed	1.58

The proof is completed. □

Theorem 3 *Given* $\epsilon, \delta \in (0, 1)$, *the Algorithm 5 returns a solution A satisfying:*

a) $\Pr[|A| \leq 1 + |A^*| \cdot \ln \frac{\gamma - \gamma\epsilon}{\epsilon}] \geq 1 - \frac{\delta}{n}$.

b) $\Pr\left(\mathbb{D}(A) \geq \gamma \cdot \frac{1-\epsilon}{1+\epsilon} - \epsilon\right) \geq 1 - \delta$.

The proof of Theorem 3 applies Lemma 7 and is similar to the proof of Theorem 2.

Complexity of ISMD ***algorithm*** Denote M is the expected running time for generating a IDS set, Algorithm 5 requires $q \frac{(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(\binom{n}{i_{max}})/\delta)$ in the worst-case and thus the complexity of Algorithm 5 is:

$$O\left(q i_{max} \rho(S) \ln(\binom{n}{i_{max}})/\delta)\epsilon^{-2})M\right)$$

The ISMD and SMD algorithms provide the same theoretical result, however, the sample complexity of ISMD is smaller than that of SMD by a factor of q, $(q < 1)$, which leads to the fact that the running time of ISMD is less than that of SMD. This observation is consistent with the experiment results on the real social networks in Sect. 5.

5 Experiment

In this section, we conduct comprehensive experiments to compare the performance of our proposed algorithms to the state-of-the-art ones on three aspects: *solution quality (size of monitor set), running time and memory usage*.

5.1 Experimental settings

Datasets For the comprehensive experimental purpose, we select a diverse set of 5 datasets with different sizes. The description of those datasets is provided in Table 2.

– ***Email-Eu-Core*** The network was generated using email data from a large European research institution. The e-mails only represent communication between institution members (the core), and the dataset does not contain incomming and outgoing messages outside the institution.

- ***Wiki-Vote*** The network contains all the Wikipedia voting data from the inception of Wikipedia until January 2008. Nodes in the network represent wikipedia users and a directed edge from node i to node j represents that user i voted on user j.
- ***CA-HepPh*** Arxiv HEP-PH (High Energy Physics - Phenomenology) collaboration network is from the e-print *arXiv.org* and covers scientific collaborations between authors' papers submitted to High Energy Physics - Phenomenology category. The data covers papers in the period from January 1993 to April 2003 (124 months).
- ***CA-AstroPh*** Arxiv ASTRO-PH (Astro Physics) collaboration network is from the e-print *arXiv.org* and covers scientific collaborations between authors' papers submitted to Astro Physics category. The data also covers papers in the period from January 1993 to April 2003 (124 months).
- ***Email-Eu-All*** The network was generated using email data from a large European research institution. For a period from October 2003 to May 2005 (18 months) they have anonymized information about all incoming and outgoing email of the research institution.

Algorithms compared We compare Greedy, SMD, and ISMD with OPIM [47], the state-of-the-art Reverse Reachable (RR) sampling algorithm for Influence Maximization problem and several common baselines for investigating information diffusion problem [7,20,42,53,54]. In baseline algorithms, we use Monte Carlo method with 10,000 times to estimate of detection function. For each algorithm, we run 10 times to get the average results. Details of these algorithms are described as follows:

- OPIM [47]: This is the current state-of-the-art algorithm that use the sample technique to solve the IM problem, where the number of seeds k is an input. Because of the similarity between DS and RR set, we use OPIM for the MBD problem in comparison with our algorithms. Besides, since our problem asks to minimize the number of monitor nodes, this algorithm cannot be applied directly. Therefore, we adapt this algorithm with some modifications by performing a binary search on k in the interval $[1, n]$. We choose this approach over starting at $k = 1$ and at each iterator of the binary search, OPIM utilizes the value of k in question until the algorithm finds the minimum k so that the estimation of the value of $\mathbb{D}$ is at least γ. There are at most $\log_2 n$ iterators.
- Degree: The heuristic algorithm based on the measurement of degree. We select nodes with the highest degree and we keep on adding the highest-degree nodes until detection function of the selection of nodes exceeds γ.
- Random: We randomly select nodes until detection function of the selection of nodes exceeds γ.
- PageRank [40]: A link analysis algorithm to rank the importance of pages in a Web graph. We implement the power method with a damping factor of 0.85 and keep on adding the highest-rank node until detection function exceeds γ.

Weight settings We use *Trivalency model* [7,20,36,55] to choose the weight of the edges. In this model, instead of assuming all nodes are equally influential, influence probabilities are drawn uniformly at random from a predetermined set of probabilities, here we used $\{0.001, 0.01, 0.1\}$. The idea of this model is that nodes whose (outgoing) influence is 0.001 can be thought of low influence nodes, with 0.01 corresponding to medium influence and 0.1 to high influence.

Table 3 Values of Υ and ϵ for each network

Dataset	$\rho(S)$	Ψ	ϵ
Email-Eu-Core	249.33	1.0	0.01
Wiki-Vote	1784.78	0.2	0.01
CA-HepPh	3009.29	0.2	0.01
CA-AstroPh	4726.66	0.4	0.1
Email-Eu-All	171,217	0.1	0.1

Parameters In all the experiments, we keep $\delta = 1/n$ as a general setting [36,47–49]. Suspected nodes are randomly chosen with the size $n/2$ and the probability $\rho(u)$ is randomly chosen in [0, 1]. We choose ϵ and γ depending on the size of the network. Denote $\Psi = \gamma/\rho(S)$ reflect the relation between γ and $\rho(S)$. The values of these parameters are described in Table 3. The running time of each algorithm is limited within 24 hours.

Environment All our experiments are carried out using a Linux machine with a 2 x Intel(R) Xeon(R) CPU E5-2697 v4 @ 2.30GHz 8 x 16 GB DIMM ECC DDR4 @ 2400MHz. Our implementation is written in C++ and compiled with GCC 4.7. We use OPENMP library for parallel programming.

5.2 Experiment results

Solution quality We first compare the quality solutions of algorithms which are measured by the size of monitor set. The results is presented in Fig. 1 in which the better algorithm returns the smaller-size monitor set. We observe that SMD and ISMD have the same performance in all cases, which outperform other algorithms by a large gap. The bigger value of Ψ is, the greater gap between our proposed algorithms and other algorithms is. Specificity, with the same value of Ψ, SMD and ISMD are up to 3.9 times better than OPIM, 2.3 times better than Greedy. Our algorithms also are several times better than baseline methods. This proves that the proposed framework algorithm for SMD and ISMD is more efficient than the other algorithms. It not only selects the smaller-size set of nodes but also ensures the approximation guarantees of the solutions. OPIM selects too many vertices because the framework of binary search may not work well in the circumstance MBD problem. In addition, the estimation of the detection function by DS and IDS concepts gives better and more efficient results than Monte-Carlo simulation method in Greedy algorithm.

Running time Figure 2 reveals the running time of the tested algorithms. In overall, both SMD and ISMD significantly outperform the rest of the algorithms in terms of running time. Our algorithms are faster than OPIM on the most of networks (up to 1.5 times faster), OPIM only gives a better time on Email-Eu-Core network. This is because OPIM takes a long time for binary search to get the good solution. Compare with Greedy, SMD is up to 10.2 times faster than Greedy and ISMD is up to 12.4 times faster than Greedy. For the large networks such as Email-Eu-All, CA-AstroPh, Greedy can not finish within limited time while SMD and ISMD algorithms still work and give good results. This indicates that the estimation of detection function by DS and IDS

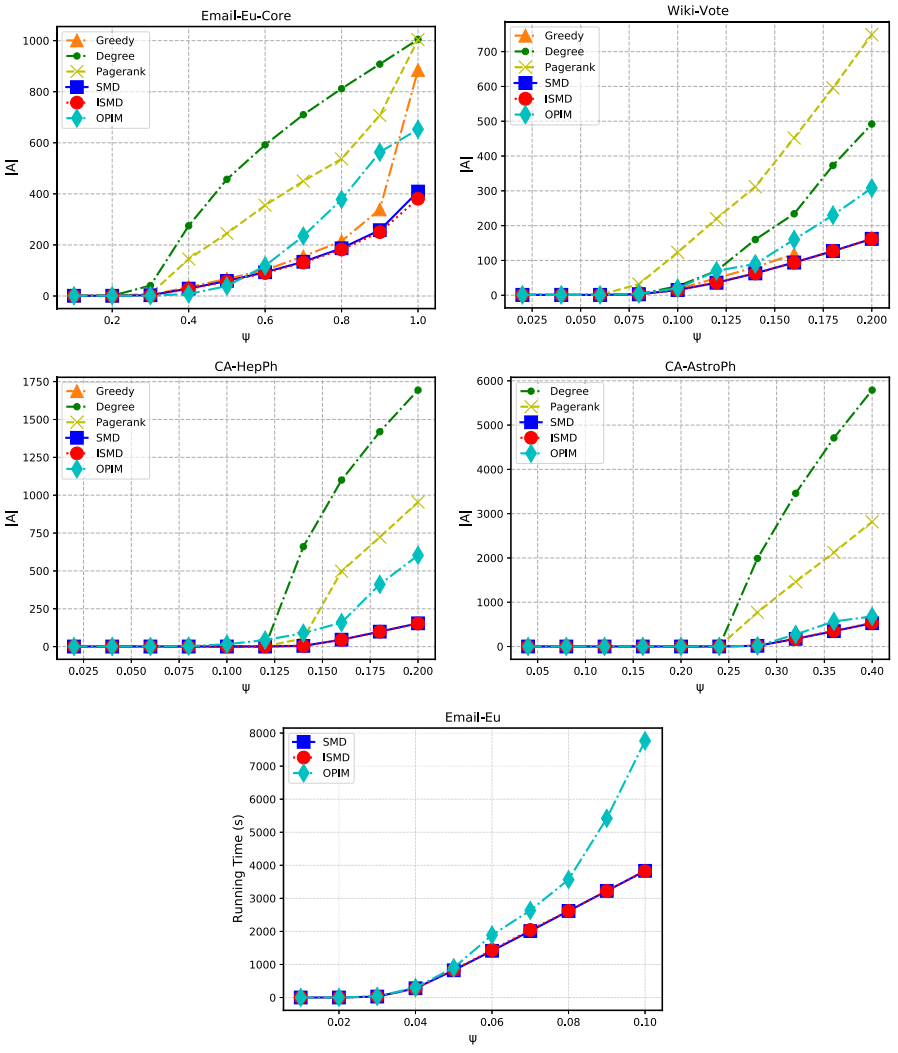

Fig. 1 Size of solutions of algorithms

is faster than using traditional Monte Carlo simulation method of Greedy. Compare SMD to ISMD, the average running time of ISMD is up to 1.4 faster than SMD. The main reason is that the number of required samples of ISMD is lower than that of SMD. Unsurprisingly, the baseline algorithms have small running time since they are simple heuristic algorithms with low complexity.

Memory usage and number of samples The results on memory usage of the SMD and ISMD algorithms are shown in Table. 4, and the number of samples generated by them is shown in the Fig. 3. We do not represent the memory usage of Greedy and baseline algorithms because their memories are small and fixed regardless of changing Ψ. SMD, ISMD and OPIM consume more memories than the other because of wasting

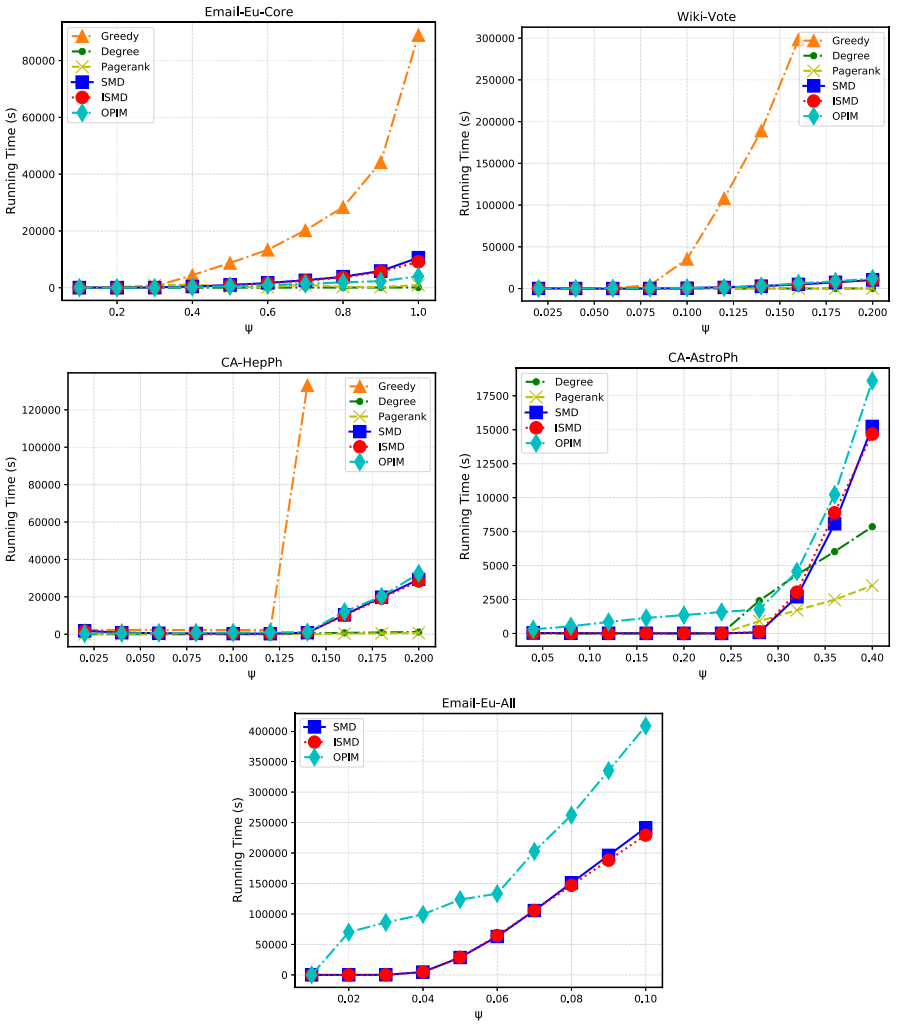

Fig. 2 Running time of the algorithms

memories for storing samples. The results show that the number of required samples and memories usage of ISMD is the smallest. The number of required samples of ISMD is up to 5.14 and 8.6 times smaller than that of SMD and OPIM, respectively. Also, these results confirm our theoretical establishment in Sect. 4 that the sample complexity of ISMD is less than that of SMD by a factor of $q < 1$. OPIM needs more samples than our algorithms because it does not reuse samples generated in previous steps. Certainly, the memory usage by ISMD is lower than that SMD of OPIM. However, the gap between SMD and ISMD is negligible. This is because of the number of nodes of an IDS samples is larger than that of a DS, so we need a more memory to store an IDS sample. This result, besides the size of monitor set and the running time, clearly shows the superiority and efficiency of ISMD compared with SMD and OPIM.

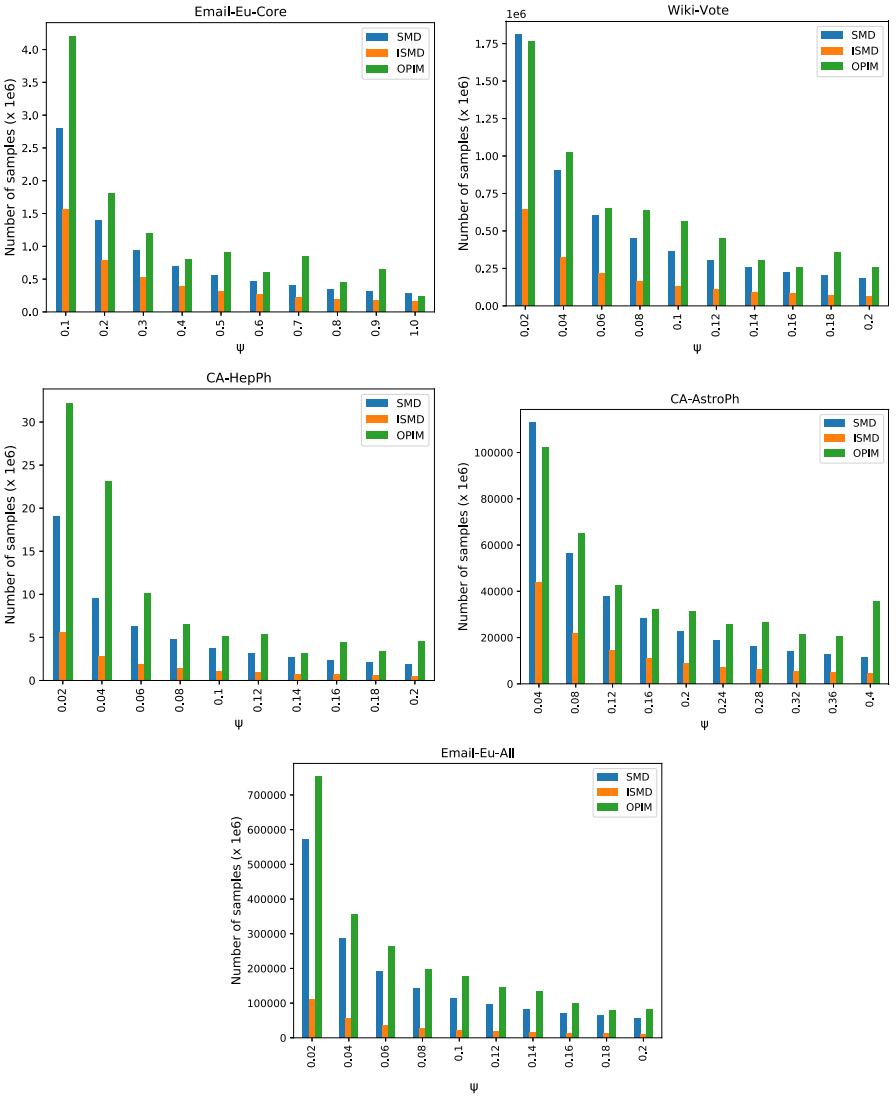

Fig. 3 Comparison number of number of samples generated by SMD, ISMD and OPIM

Table 4 Memory usage ($\times$ 1000 MB) of SMD, ISMD and OPIM

Algo.	Email-Eu-Core ($\Psi = 1$)	Wiki-Vote ($\Psi = 0.2$)	CA-Hep. ($\Psi = 0.2$)	CA-Astro. ($\Psi = 0.4$)	Email-Eu-All ($\Psi = 0.1$)
SMD	14.2	41.3	28.8	14.7	25.5
ISMD	13.7	37.2	28.4	13.9	25.4
OPIM	13.9	52.6	62.3	35.1	30.2

6 Conclusion

In this paper, we propose MBD problem which aims at finding the smallest set of nodes to place monitors in a social network to detect misinformation from suspected nodes so that the expected detection function is greater than or equal to a threshold $\gamma > 0$. Besides showing challenge for solving MBD, we propose three algorithms including: Greedy, SMD and ISMD, in which SMD and ISMD are randomized approximation algorithms that outperform other algorithms. In the future, we will further improve the running time of these algorithms making them applicable to billion-scale networks.

Acknowledgements The second and fourth authors were supported by the Vietnam Academy of Science and Technology under grant VAST01.05/21-22.

References

1. (2018) Misinformation on social media led to pune violence: Minister. In: https://www.ndtv.com/mumbai-news/misinformation-on-social-media-led-to-pune-violence-minister-1795562. Accessed 22 Nov 2020
2. Aslay, Ç., Barbieri, N., Bonchi, F., Baeza-Yates, R.A.: Online topic-aware influence maximization queries. In: Proceedings of the 17th International Conference on Extending Database Technology, EDBT 2014, Athens, Greece, March 24–28, 2014, pp. 295–306 (2014)
3. Borgs, C., Brautbar, M., Chayes, J.T., Lucier, B.: Maximizing social influence in nearly optimal time. In: Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5–7, 2014, pp. 946–957 (2014)
4. Budak, C., Agrawal, D., El Abbadi, A.: Limiting the spread of misinformation in social networks. In: Proceedings of the 20th International Conference on World Wide Web, WWW 2011, Hyderabad, India, March 28–April 1, 2011, pp. 665–674 (2011)
5. Castillo, C., Mendoza, M., Poblete, B.: Information credibility on twitter. In: Proceedings of the 20th International Conference on World Wide Web, WWW 2011, Hyderabad, India, March 28–April 1, 2011, pp. 675–684 (2011)
6. Chen, T., Li, X., Yin, H., Zhang, J.: Call attention to rumors: Deep attention based recurrent neural networks for early rumor detection. In: Trends and Applications in Knowledge Discovery and Data Mining—PAKDD 2018 Workshops, BDASC, BDM, ML4Cyber, PAISI, DaMEMO, Melbourne, VIC, Australia, June 3, 2018, Revised Selected Papers, pp. 40–52 (2018)
7. Chen, W., Wang, C., Wang, Y.: Scalable influence maximization for prevalent viral marketing in large-scale social networks. In: Rao, B., Krishnapuram, B., Tomkins, A., Yang, Q. (eds) Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, July 25–28, 2010, ACM, pp. 1029–1038 (2010a)
8. Chen, W., Yuan, Y., Zhang, L.: Scalable influence maximization in social networks under the linear threshold model. In: ICDM 2010, The 10th IEEE International Conference on Data Mining, Sydney, Australia, 14–17 December 2010, pp. 88–97 (2010b)
9. Chen, W., Lu, W., Zhang, N.: Time-critical influence maximization in social networks with time-delayed diffusion process. In: Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22–26: Toronto. Ontario, Canada (2012)
10. Christakis, N.A., Fowler, J.H., Sporns, O.: Social network sensors for early detection of contagious outbreaks. PLoS ONE **5**, 9 (2010)
11. Chung, F.R.K., Lu, L.: Survey: concentration inequalities and martingale inequalities–a survey. Int. Math. **3**(1), 79–127 (2006)
12. Cui, P., Jin, S., Yu, L., Wang, F., Zhu, W., Yang, S.: Cascading outbreak prediction in networks: a data-driven approach. In: The 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2013, Chicago, IL, USA, August 11–14, 2013, pp. 901–909 (2013)
13. Domm, P.: False rumor of explosion at white house causes stocks to briefly plunge; ap confirms its twitter feed was hacked (2013). https://www.cnbc.com/id/100646197. Accessed 22 Nov 2020

14. Du, N., Song, L., Gomez-Rodriguez, M., Zha, H.: Scalable influence estimation in continuous-time diffusion networks. In: Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems. Proceedings of a Meeting Held December 5–8, 2013, pp. 3147–3155. Lake Tahoe, Nevada, United States (2013)
15. Fan, J., Qiu, J., Li, Y., Meng, Q., Zhang, D., Li, G., Tan, K., Du, X.: OCTOPUS: an online topic-aware influence analysis system for social networks. In: 34th IEEE International Conference on Data Engineering, ICDE 2018, Paris, France, April 16–19, 2018, pp. 1569–1572 (2018)
16. Feige, U.: A threshold of ln n for approximating set cover. J. ACM **45**(4), 634–652 (1998)
17. Gomez-Rodriguez, M., Song, L., Du, N., Zha, H., Schölkopf, B.: Influence estimation and maximization in continuous-time diffusion networks. ACM Trans. Inf. Syst. **34**(2), 9:1-9:33 (2016)
18. Goyal, A., Bonchi, F., Lakshmanan, L.V.S., Venkatasubramanian, S.: On minimizing budget and time in influence propagation over social networks. Soc. Netw. Anal. Min. **3**(2), 179–192 (2013)
19. He, X., Song, G., Chen, W., Jiang, Q.: Influence blocking maximization in social networks under the competitive linear threshold model. In: Proceedings of the Twelfth SIAM International Conference on Data Mining, Anaheim, California, USA, April 26–28, 2012., pp. 463–474 (2012)
20. Kempe, D., Kleinberg, J.M., Tardos, É.: Maximizing the spread of influence through a social network. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 24–27, 2003, pp. 137–146 (2003)
21. Khalil, E.B., Dilkina, B.N., Song, L.: Scalable diffusion-aware optimization of network topology. In: The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD'14, New York, NY, USA—August 24–27, 2014, pp. 1226–1235 (2014)
22. Krause, A., Guestrin, C.: Optimizing sensing: from water to the web. IEEE Comput. **42**(8), 38–45 (2009)
23. Kwon, S., Cha, M., Jung, K., Chen, W., Wang, Y.: Prominent features of rumor propagation in online social media. In: 2013 IEEE 13th International Conference on Data Mining, Dallas, TX, USA, December 7–10, 2013, pp. 1103–1108 (2013)
24. Leskovec, J., Kleinberg, J.M., Faloutsos, C.: Graph evolution: densification and shrinking diameters. TKDD **1**(1), 2 (2007a)
25. Leskovec, J., Krause, A., Guestrin, C., Faloutsos, C., VanBriesen, J.M., Glance, N.S.: Cost-effective outbreak detection in networks. In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Jose, California, USA, August 12–15, 2007, pp. 420–429 (2007b)
26. Leskovec, J., Huttenlocher, D.P., Kleinberg, J.M.: Predicting positive and negative links in online social networks. In: Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26–30, 2010, pp. 641–650 (2010a)
27. Leskovec, J., Huttenlocher, D.P., Kleinberg, J.M.: Signed networks in social media. In: Proceedings of the 28th International Conference on Human Factors in Computing Systems, CHI 2010, Atlanta, Georgia, USA, April 10–15, 2010, pp. 1361–1370 (2010b)
28. Li, G., Chen, S., Feng, J., lee Tan, K., WSL.: Efficient location-aware influence maximization. In: 34th IEEE International Conference on Data Engineering, ICDE 2018, Paris, France, April 16–19, 2018, pp. 1569–1572 (2018)
29. Li, Y., Zhang, D., Tan, K.: Targeted influence maximization for online advertisements. PVLDB **8**(10), 1070–1081 (2015)
30. Luckerson, V.: Fear, misinformation, and social media complicate ebola fight (2014). http://time.com/3479254/ebola-social-media/. Accessed 22 Nov 2020
31. Ma, J., Gao, W., Wei, Z., Lu, Y., Wong, K.: Detect rumors using time series of social context information on microblogging websites. In: Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015, Melbourne, VIC, Australia, October 19–23, 2015, pp. 1751–1754 (2015)
32. Ma, J., Gao, W., Mitra, P., Kwon, S., Jansen, B.J., Wong, K., Cha, M.: Detecting rumors from microblogs with recurrent neural networks. In: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9–15 July 2016, pp. 3818–3824 (2016)
33. Ma, J., Gao, W., Wong, K.: Detect rumors in microblog posts using propagation structure via kernel learning. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30–August 4, Volume 1: Long Papers, pp. 708–717 (2017)

34. Ma, J., Gao, W., Wong, K.: Detect rumor and stance jointly by neural multi-task learning. In: Companion of the The Web Conference 2018 on The Web Conference 2018, WWW 2018, Lyon , France, April 23–27, 2018, pp. 585–593 (2018a)
35. Ma, J., Gao, W., Wong, K.: Rumor detection on twitter with tree-structured recursive neural networks. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15–20, 2018, Volume 1: Long Papers, pp. 1980–1989 (2018b)
36. Nguyen, H.T., Thai, M.T., Dinh, T.N.: Stop-and-stare: Optimal sampling algorithms for viral marketing in billion-scale networks. In: Proceedings of the 2016 International Conference on Management of Data, SIGMOD Conference 2016, San Francisco, CA, USA, June 26–July 01, 2016, pp. 695–710 (2016)
37. Nguyen, H.T., Thai, M.T., Dinh, T.N.: A billion-scale approximation algorithm for maximizing benefit in viral marketing. IEEE/ACM Trans. Netw. **25**(4), 2419–2429 (2017)
38. Nguyen, H.T., Cano, A., Vu, T., Dinh, T.N.: Blocking self-avoiding walks stops cyber-epidemics: a scalable gpu-based approach. IEEE Trans. Knowl. Data Eng. **32**(7), 1263–1275 (2020)
39. Nguyen, N.P., Yan, G., Thai, M.T.: Analysis of misinformation containment in online social networks. Comput. Netw. **57**(10), 2133–2146 (2013)
40. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, http://ilpubs.stanford.edu:8090/422/, previous number = SIDL-WP-1999-0120 (1999)
41. Pham, C.V., Dinh, H.M., Nguyen, H.D., Dang, H.T., Hoang, H.X.: Limiting the spread of epidemics within time constraint on online social networks. In: Proceedings of the Eighth International Symposium on Information and Communication Technology, Nha Trang City, Viet Nam, December 7–8, 2017, pp. 262–269 (2017)
42. Pham, C.V., Thai, M.T., Duong, H.V., Bui, B.Q., Hoang, H.X.: Maximizing misinformation restriction within time and budget constraints. J. Comb. Optim. **35**(4), 1202–1240 (2018)
43. Pham, C.V., Duong, H.V., Hoang, H.X., Thai, M.T.: Competitive influence maximization within time and budget constraints in online social networks: an algorithmic approach. Appl. Sci. **9**, 11 (2019)
44. Qazvinian, V., Rosengren, E., Radev, D.R., Mei, Q.: Rumor has it: Identifying misinformation in microblogs. In: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, EMNLP 2011, 27–31 July 2011, John McIntyre Conference Centre, Edinburgh, UK, A meeting of SIGDAT, a Special Interest Group of the ACL, pp. 1589–1599 (2011)
45. Ruchansky, N., Seo, S., Liu, Y.: CSI: A hybrid deep model for fake news detection. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Singapore, November 06–10, 2017, pp. 797–806 (2017)
46. Smith, K.: Marketing: 115 amazing social media statistics and facts. In: https://www.brandwatch.com/blog/96-amazing-social-media-statistics-and-facts/ (2018)
47. Tang, J., Tang, X., Xiao, X., Yuan, J.: Online processing algorithms for influence maximization. In: Das G, Jermaine CM, Bernstein PA (eds) Proceedings of the 2018 International Conference on Management of Data, SIGMOD Conference 2018, Houston, TX, USA, June 10–15, 2018, ACM, pp. 991–1005 (2018)
48. Tang, Y., Xiao, X., Shi, Y.: Influence maximization: near-optimal time complexity meets practical efficiency. In: International Conference on Management of Data, SIGMOD 2014, Snowbird, UT, USA, June 22–27, 2014, pp. 75–86 (2014)
49. Tang, Y., Shi, Y., Xiao, X.: Influence maximization in near-linear time: A martingale approach. In: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, Melbourne, Victoria, Australia, May 31–June 4, 2015, pp. 1539–1554 (2015)
50. Wang, X., Zhang, Y., Zhang, W., Lin, X.: Efficient distance-aware influence maximization in geo-social networks. IEEE Trans. Knowl. Data Eng. **29**(3), 599–612 (2017)
51. Yin, H., Benson, A.R., Leskovec, J., Gleich, D.F.: Local higher-order graph clustering. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13–17, 2017, pp. 555–564 (2017)
52. Zhang, H., Alim, M.A., Thai, M.T., Nguyen, H.T.: Monitor placement to timely detect misinformation in online social networks. In: 2015 IEEE International Conference on Communications, ICC 2015, London, United Kingdom, June 8–12, 2015, pp. 1152–1157 (2015)
53. Zhang, H., Alim, M.A., Li, X., Thai, M.T., Nguyen, H.T.: Misinformation in online social networks: detect them all with a limited budget. ACM Trans. Inf. Syst. **34**(3), 18:1-18:24 (2016a)

54. Zhang, H., Kuhnle, A., Zhang, H., Thai, M.T.: Detecting misinformation in online social networks before it is too late. In: 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2016, San Francisco, CA, USA, August 18–21, 2016, pp. 541–548 (2016b)
55. Zhang, Y., Prakash, B.A.: Data-aware vaccine allocation over large networks. ACM Trans. Knowl. Discov. Data **10**(2), 20:1-20:32 (2015)